WHAT'S NEW

at the Joint Integration Test Facility

JITF - What's New

- NT Logo Testing

- Code Analysis

- Test Report Format

- JITF Automated Test Tools

Logo Testing

- Procedures developed by Veritest
  - Microsoft authorized NT Logo tester for commercial applications

- JITF personnel trained in Veritest process
  - Procedures are very similar to existing DODIIS Integration Test Procedures
    - DODIIS Procedures go farther/deeper into integration issues
  - JITF has incorporated NT Logo procedures into JITF Test Procedures

- JITF is the <u>only</u> organization trained to conduct this testing for classified Government developed software

Code Analysis

- OASD C3I tasked JITF to conduct Y2K Independent Verification & Validation

- Analyzed source code and identified potential problems for 11 programs

  - AMHS, Broadsword, CSP, CSE-SS, DMFE, IESS, IPL, IDEX II, ISSE Guard, SPG and TAS

  - Used automated tool to conduct code scanning

  - Potential problems forwarded to the Program Office for further analysis and resolution

Code Analysis Results

- Identified significant Y2K problem

  - Program had independently identified the problem and was working correction

  - Confirmed that the programs conducted thorough test and preparation for Y2K

- Dates used as file names - not used in calculations

- Lines of unused code contained invalid dates

- Unused code creates potential problems that are not Y2K related

  - Potential configuration management issues
  - Foreign disclosure to allies and partners
  - Information Assurance
  - Quality Assurance

Test Report

- JITF updating Test Report Format

  - Streamlines report generation
  - Improves readability
  - Provides clear recommendation

- Pass/Fail criteria is clearly defined

  - Seventy (70%) percent of Integration Requirements must be met
  - No impact code 1s can be open

- Metrics incorporated into report to graphically display results for better understanding

Test Report Format

- Section 1 - Overview
  - Paragraph 1.4 provides summary and test metrics
  - Pass/Fail information is located here

- Section 2 - Configuration Information

- Section 3 - Detailed Analysis
  - Identifies all findings and categorizes according to recommendation

- Section 4 - Bibliography

- Section 5 - Participants

- Section 6 - Acronyms

Test Report Format

- Section 7 - Defines Impact Code Levels

- Section 8 - Integration Requirements Matrix

- Section 9 - Y2K Integration Matrix

- Section 10 - Open Software Findings

- Section 11 - Software Findings Closed with Test

- Section 12 - Open Document Findings

- Section 13 - Document Findings Closed with Test

Test Report Metrics

- Figure 1-1 Integration Requirements Met and Unmet for the application and version under test

  – For applicable requirements
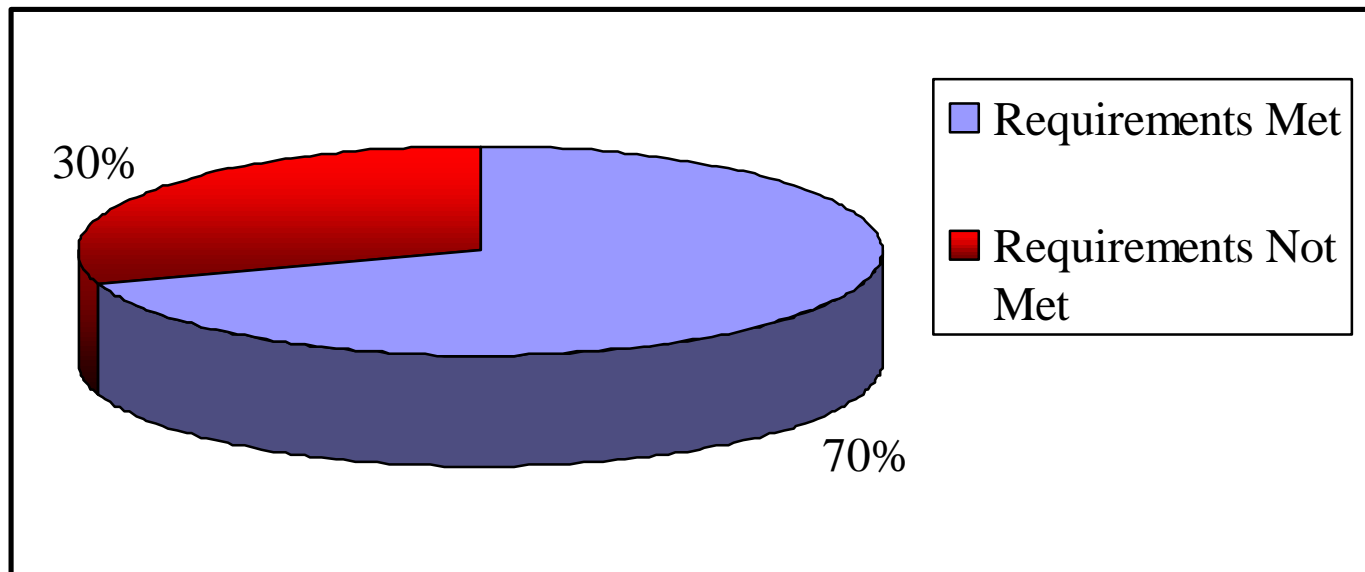
Test Report Metrics

- Figure 1-2 Software Findings Statistics

    - Identifies new software findings for version under test by impact code
    - Identifies software finding closed during the test
        - Findings identified in previous versions of the software
        - May include findings closed during a "re-test" of the current version under test
    - Identifies the total number of open software findings for the application
        - Includes findings identified in previously tested versions that remain open
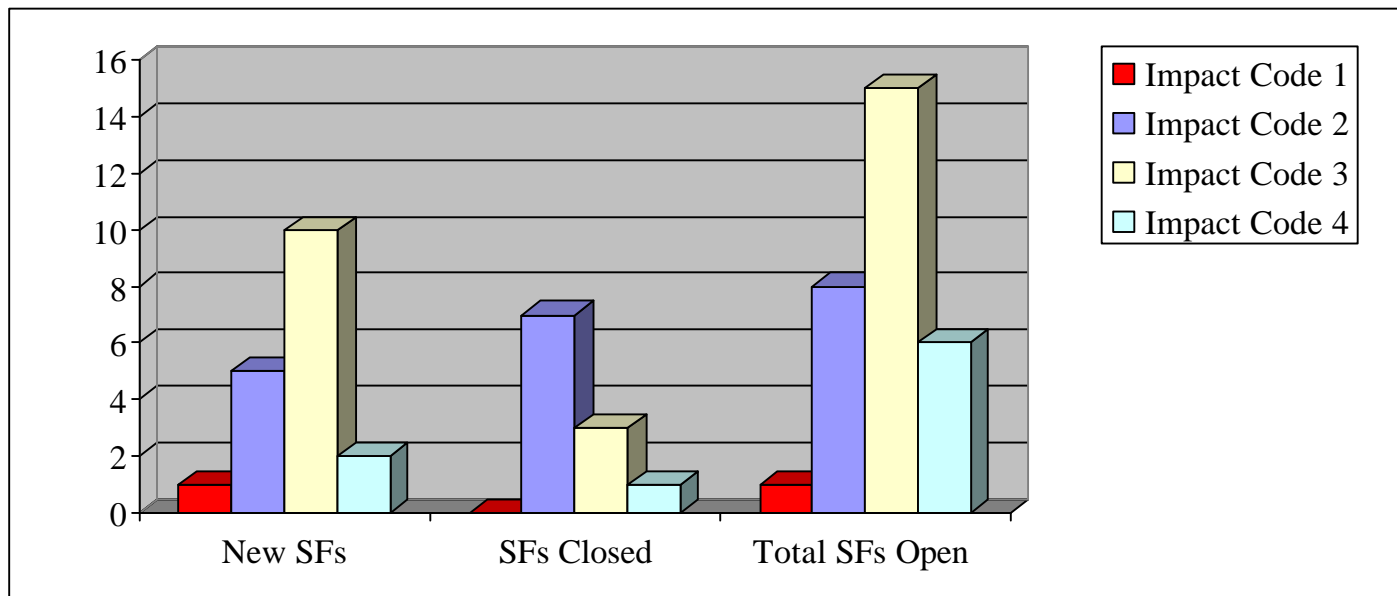
Test Report Metrics

- Figure 1-3 Document Findings Statistics
  - Same information as provided for Software Findings
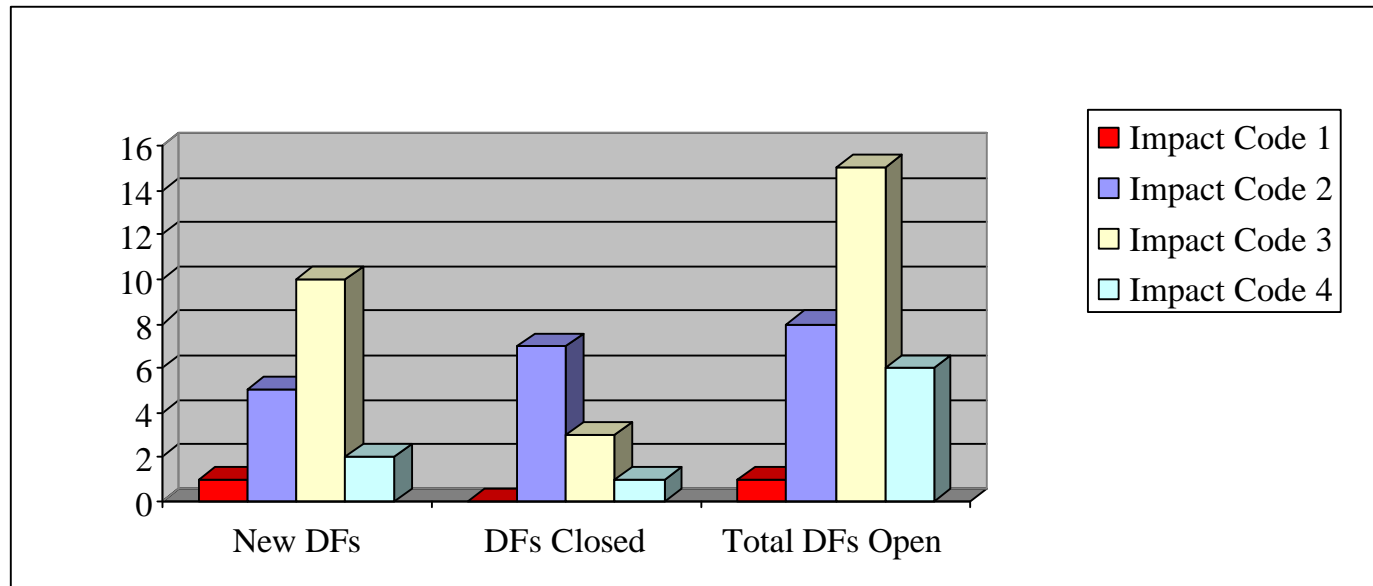
JITF Test Tools

- JITF Automated Tool (JAT)

  – Perl scripts with GUI front-end (UNIX)

  – Automatically evaluates several requirements

  – Minimizes manual evaluation

  – Provides Pass/Fail report for tested requirements

  – Identifies system deltas via pre-IMA installation vs post-IMA installation "snapshots"

JITF Test Tools

- NoteTaker

  - Microsoft Access forms/database for tracking all aspects of JITF IMA evaluation

    - Compile internal notes used to document test event

    - Generate real-time Software/Document Findings

    - Generate Requirements matrix

  - Reduces time to generate draft test report

JITF Test Tools (continued)

- Future Development
  - JAT-NT
    - Port requirement evaluation tool to NT
  - JAT-Trusted Solaris
    - Specific for Trusted Solaris environment
  - Truss
    - Create GUI for UNIX system call tracing tool
    - Assist in IMA evaluation/troubleshooting

- Demonstrations of these tools are available during the TPOC